

Design and implementation of a function block based holonic control architecture for a new generation flexible manufacturing system

Abstract

In this research work a control architecture which gives response to the requirements of new generation of flexible manufacturing systems in terms of flexibility, reconfigurability, robustness and autonomy is designed and implemented. To do so the main principles of the Holonic Manufacturing paradigm are applied using the IEC61499 function block (FB) technology. Unlike other similar research proposals, in this work FBs are not relegated to low-level control but are used to model manufacturing execution and control high-level control tasks. This is done with the objective of evaluating the viability of using FBs to develop holonic architectures in comparison to more established technologies like multi-agent systems. Moreover, the proposed control architecture also focuses on better integrating and exploiting the products' information to enhance its flexibility and adaptability. For this STEP-NC (ISO14649) is used to model richer process plans which include manufacturing alternatives and could be easily integrated in the control itself.

Keywords: Intelligent Manufacturing System; Manufacturing Control; Holonic Architecture; Function Blocks; STEP-NC Process Plans

1. Introduction

In last the decades the global demand of products has suffered an important evolution. From a steady demand of a small assortment of products it has evolved to a volatile demand of an enormous variety of products, resulting in the apparition of new production paradigms (Koren, 2013): the mass customization, that has led the evolution of manufacturing in the last decades, and more recently the mass individualization.

These emerging production paradigms have forced the manufacturing automation to face significant challenges over the twenty past years, with the objective of building the next generation of manufacturing systems. These challenges brought to the appearance of a new operational paradigm: the Intelligent Manufacturing Systems (IMS) (Yoshikawa, 1995; Valckenaers, 1998; 2000). According to Morel et al. (2003) it is a new systemic paradigm, that is, the whole manufacturing enterprise is analyzed as a system, that goes beyond the Integrated Manufacturing (IM) paradigm (Vernadat, 1996), which cannot neither face the problems related to the interface between the enterprise corporate and the manufacturing shop floor

level nor the dynamic loops that are produced in that interface and that require moving from an the hierarchical IM to the heterarchical IMS paradigm. One of the points that IMS paradigm considers is to organize human resources and machines into a networked and evolving system that adapts itself to the requirements of agile manufacturing. Manufacturing systems developed under their principles must distribute the digital intelligence across the field factory in order to enable flexible and autonomous operation of distributed units to transform information flows into products flows (Morel et al., 2003).

Around this vision two technologies have recently generated a considerable amount of attention: Multi-Agents Systems (MAS) and Holonic Manufacturing Systems (HMS), until the point that are considered key information/technological approaches to meet the dynamic requirements of agile manufacturing (Monostori, 2003). These technologies were born in different research fields, MAS appeared in the distributed artificial intelligence (DAI) field while HMS were originated in the manufacturing field, both share similar concepts and have shown promising results. Both, Agents and Holons, have provided a new abstraction metaphor for designing highly distributed and intelligent industrial control systems featuring attributes like autonomy, robustness, survivability, adaptation, reconfiguration, cooperation and openness (Vrba et al., 2011; HMS, 1994). According to Giret and Botti (2004) a manufacturing holon is an autonomous and cooperative building block of a manufacturing system for transforming, transporting, storing and/or validating information and/or physical objects. Therefore, a manufacturing holon always contains an information processing part and, optionally, a physical device. This distinction between information and physical processing is taken into account in most of HMS architectures. For information processing most of the authors consider MAS as the best fitting technology, since it can provide the necessary reasoning techniques for holons to process the information and cooperate, so they can easily interact between them forming holarchies. However, for the physical processing many authors, senior members of HMS consortium among them, have proposed IEC61499 function block standard (IEC, 2005) for developing the low-level control architectures (Balasubramanian, 2001; Christensen, 2002; 2003), because of its capabilities in terms of dynamic reconfiguration, safety management and system validation (Brennan, 2007). In the same line, in (Fletcher et al, 2000 and Vrba et al. 2011) authors consider suitable the use of FBs for low-level real-time process/machine interaction. In details, the latter think over how to simplify the transition from the classical “old-fashioned” approaches to “novel” control paradigms based on the use of MAS, showing that it is

Author.

necessary to consider the capability of agents to interface with the physical equipment on the shop floor. In this line they propose the design of a coupled architecture (Agents and FB's) referred to as physical agent or holonic agent in their terminology. It consists of a component that is an encapsulation of a high-level control part (HLC), a low-level control subsystem (LLC) and the control interface for interactions between both HLC and LLC. For the LLC module these authors propose using the IEC 61131-3 or IEC 61499 standards, while the HLC module is embodied by the "agent" in the pure MAS sense.

The characteristics of the IEC 61499 FB based architectures for handling the low-level control in an intelligent and distributed manner makes them suitable for the development of the next generation flexible manufacturing systems (FMS). In order to the FMSs could give response to the requirements of massive individualized demand, these have to overcome some weakness that the current FMS users recognize (Mehrabi et al, 2008), especially in terms of reconfigurability and initial cost. Endowing the system with reconfigurability capacity would allow avoiding initial oversizing of the FMS reducing their cost, since it would allow adding new soft and hard components only when they are needed.

In this context described above, in this research project it was proposed the idea of using function blocks to cover Manufacturing Operations Management tasks (ISA95 level 3, Figure 1) of a FMS composed of general purpose CNC machines, where other authors propose the use of "pure" software agents. Therefore, our approach proposes to promote from the FBs original low-level domain (ISA95 level 1) to higher levels control tasks (ISA-95 levels 2 and 3).

This promotion of functions blocks has 2 goals: improve the reconfigurability of FMSs and exploring the viability of using FBs as technology to implement HMS. Additionally, this research also searches improving the flexibility and adaptability of the FMSs through the integration of design and process planning information into control tasks. It is achieved using STEP-NC (ISO14649) models so the control itself can exploit information such as process plans with alternatives.

Design and implementation of a function block based holonic control architecture for a new generation flexible manufacturing system

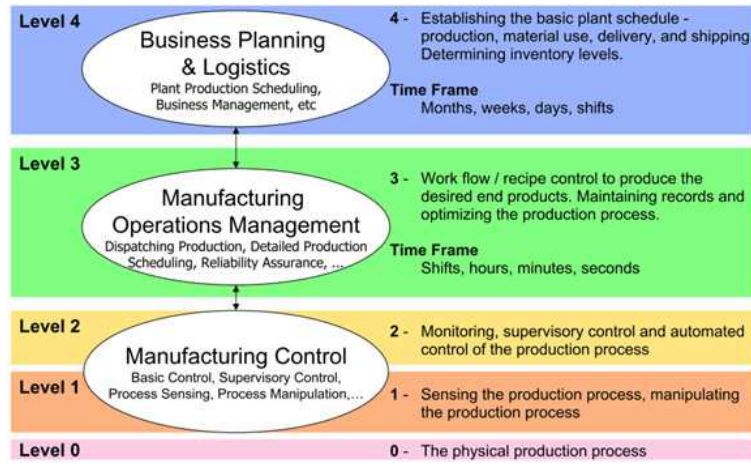


Figure 1: Enterprise functional levels according to AINSI/ISA-95 standard.

The rest of the document is structured as follows: In the first section, the enabling technologies of IEC61499 function blocks and the STEP-NC process plans will be introduced. Following the proposed holonic control architecture will be detailed and the proposed holons will be analysed in depth. Then the benefits and drawbacks found when using functions blocks to model high-level control tasks will be highlighted. Finally, we will present some conclusions.

2. Enabler technologies

The proposed architecture is based on two key technologies: IEC 61499 function blocks, to design and implement a distributed control architecture based on the holonic approach, and STEP-NC standard, to model non-linear process plans. Below both of these technologies are presented and their advantages highlighted.

2.1 IEC61499 function blocks overview

The IEC61499 standard is a reference architecture created to facilitate the development of distributed control applications (Vyatkin, 2013). This standard is built around the concept of Functional Block (FB), which represents the smallest logic unit in the control system. FBs can be seen as the traditional blocks used in the Block Diagrams; they are individual units composed of two parts: their interface and their internal behaviour. The block's interface determines how an FB will communicate and interact with other FBs, while the internal behaviour determines how the block will respond to a specific input. There are three types of FBs: Basic (BFB), Composite (CFB) and Service Interface (SIFB). These three types of blocks are combined and interconnected to form FBs networks, which can be used to model and implement entire control systems.

Author.

BFBs are the elemental blocks of the control applications: they are indivisible and autonomous, in the sense that they do not need anything else to carry out their internal functions. CFBs are blocks that work as containers of networks of other blocks. SIFBs belong to a special type of block whose interface is specified but whose internal behaviour is only defined in its implementation, therefore remaining outside the scope of the IEC61499. All these types share the same interface, but have different internal definitions. For the sake of brevity only the BFB will be detailed, since it is the most important type.

Figure 2 shows an example of an FB interface; it is composed of its input and output events and its data connections. The events are used to trigger the execution of the FBs, since the IEC61499 architecture is based on an event-driven execution model, where blocks are only executed when required, remaining idle the rest of the time, in contrast to other techniques, like IEC61131, where the control units are executed periodically. The data connections are associated with the events and used to transmit information between blocks at execution time.

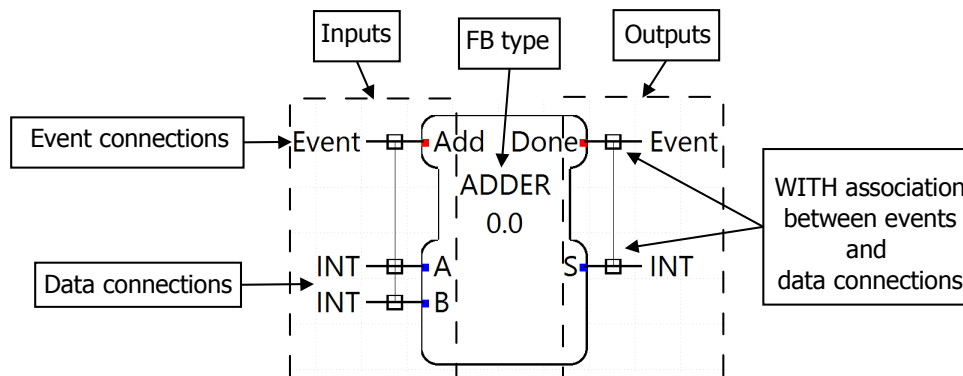


Figure 2.Example of functional block interface

Internally, BFBs consist of a set of algorithms, a context of variables, which can only be utilized by the algorithms of the block, and an execution control chart (ECC). This last element is a kind of machine-state diagram, which determines the current state of the block and how it will react when a new event arrives. Algorithms are associated with the states of the ECC, so whenever the state of the block changes, the associated algorithms are executed, finally producing one or more output events that propagate the execution between blocks.

Once the block types have been modelled, they are interconnected forming networks, which are aggregated into control applications. These networks are distributable, which means that each FB can be executed in a different resource while being part of the global system. The execution of the FB is carried out by an execution runtime which is not defined by the IEC61499

itself. This abstraction of the models respect with its implementation greatly enhances the reusability and portability of control applications and FBs, since they can be executed in any device as long as it can run an IEC61499 runtime.

Runtimes are in charge of executing the IEC61499 models, acting as an intermediate layer between the abstract model and the operating system or firmware of the device, which controls it physically. Runtimes are capable of interpreting IEC61499 models, made according to the standard, and at the same time they can utilize and access the device's interfaces such as networks, I/O, memory, and so on, since each runtime is designed specifically for a device.

2.2 STEP-NC (ISO14649) process plans

Aside of adopting a new control architecture, to increase the autonomy and adaptability of the system, that is, its intelligent behaviour, it is necessary to improve the information flow between the control and products' design and process planning process. A first and fundamental step in this way is the integration of the products' process plans into the manufacturing control. Process plans contain all the information needed to transform a raw material or semi-finished product into a finished one. Among other information, it may contain the sequence of operations to be followed, the fixtures to be used or the tools needed to carry out these operations. Process plans are generated from products' designs, a CAD design typically, by specialist manufacturing engineers or expert software capable of recognizing geometries and deciding the best manufacturing solution. When these process plans are generated they may contain a lot of alternatives to obtain the same product. On traditional manufacturing systems, these alternatives are truncated and the best solution is chosen resulting in a fixed sequence of manufacturing operations. This "best solution" is decided in function of the expected state of the manufacturing resources (available machines and tools) at the manufacturing time. However, in many cases the real conditions may be very different to the expected ones due to machines break downs, workers absenteeism or missing tools. In these cases, the production must be stopped or delayed even though there are other viable alternatives. To solve this drawback and optimize the resource usage another process planning approach appeared, namely non-linear process plans. These plans contain routes with alternatives, using AND and OR logic operators. This way, more information about the process planning process is retained and, if it is integrated with the control system, greatly enhances resource utilization. For example, if manufacturing control detects machines' failures

Author.

or delays, it can reinterpret the process plan of the parts to optimize the overall system performance.

The main problem with this approach is the modelling of these non-linearities and the technical integration of the process plans into the control architectures. These two issues can be solved by adopting an approach based on the STEP-NC (ISO 14649) standard, which is defined as a new model of data transfer between CAD/CAM systems and CNC machines based on, and harmonized with, the models for data product standardization defined by the STEP initiative. STEP-NC proposes a new way to model the process plans based on the widely known manufacturing features of STEP AP 224. This feature-based modelling allows easily creating non-linear process plans by including selective and parallel clauses.

Additionally, STEP-NC process plans features not only include information about how to execute the manufacturing operation but also enclose information about the geometry of the part and the features themselves. This geometric information can be very useful to give an additional intelligence and autonomy to the manufacturing systems. This information can be used for example to optimize the tools' paths based on the instantaneous wear of the tools or to recalculate the paths based on the measured results of previous operations. Definitely, it gives the machines' CNCs a very useful information that can boost its performance when exploited.

Finally, STEP-NC is not a simple description language for process plans but a set of standardized and structured models conceived according to the object-oriented paradigm. This structured information makes it easier to integrate the process plans into the control architecture, especially in functional blocks architecture that it is already object-oriented. For example, features of AP 224 (information objects) can be directly mapped as FBs.

3. Proposed holonic architecture

The proposed architecture itself is divided in four functional levels which cover from manufacturing orders reception at level 3, to physical interfacing with machines at level 0, as depicted in figure 2 left. The architecture is modelled entirely using IEC61499 function blocks following the guidelines proposed by Lewis (2001). All the data connections between FBs are typed according to IEC61131-3 primitive types (Integer, Boolean, Real, etc.), since although IEC61499 does not limit the creation of new data types, standardized types are preferred to improve reusability and interoperability.

Design and implementation of a function block based holonic control architecture for a new generation flexible manufacturing system

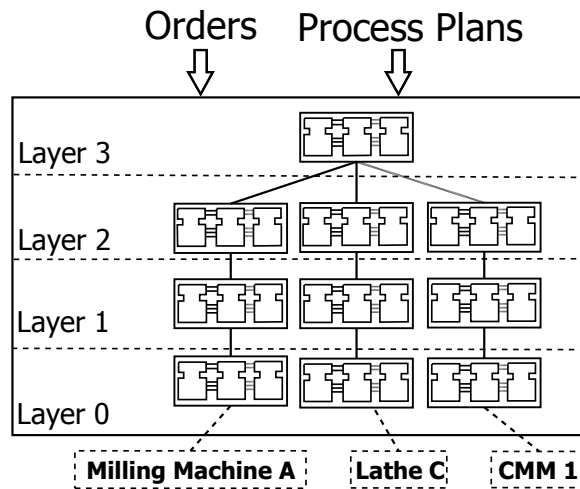


Figure 3. Proposed holonic architecture organized by functional layers.

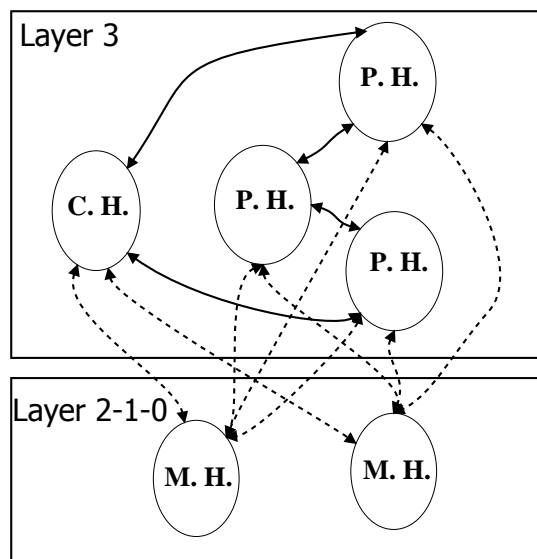


Figure 4. Proposed holonic architecture organized by holons.

As a result of exclusively using IEC61499 FBs the resulting architecture is fully distributable, that is, each of its forming parts can be executed in a different physical device without losing functionalities and without needing to redesign the control itself. Additionally, since FBs are thought to be executed through runtimes, the architecture can be classified as platform-independent. This means that physical devices of different types and vendors can be used (together) to implement the control. Typical devices capable of running IEC61499 models range from industrial computers

Author.

running Microsoft Windows or Linux Operative systems, to IEC61499 compatible and/or Linux based PLCs, or even single board computers and micro-controller embedded systems.

The main control tasks covered by the developed architecture are summarized next, classified by functional layers.

- Layer 3: Manufacturing orders reception and scheduling, part dispatching and information routing.
- Layer 2: Resources monitoring (general state, tools' availability and wear, fixtures, etc.), historic operation results and non-linear process plans handling.
- Level 1: Resource specific code generation (post-processing) and adaptive controls.
- Level 0: Physical communication with resources (CNCs and auxiliary devices).

According to its functional layers structure the proposed architecture may seem like a traditional hierarchical control architecture, where control is organised in nodes of increasing abstraction and complexity. However, the proposed architecture has a completely different internal behaviour. From an holonic point of view the architecture is composed of three types of holons: *Coordinator Holon* (CH), *Product Holon* (PH) and *Machine Holon* (MH) as depicted in figure 3 right. These holons are modelled as FBNs and, according to the criteria proposed by Giret and Botti (2004), can be classified as holons since they are: a) Autonomous, each holon can carry its own actions without needed the others; b) Cooperative, they communicate using well-defined interfaces; c) Reconfigurable, they can be added, removed or rearranged from the architecture without altering it.

Next, the design and duties of each of these holons will be detailed. For an analysis of the architecture from a functional point of view refer to (Querol et al., 2015)

Product Holon

Product Holon is defined in a similar manner than in other HMS works like PROSA (Brussel et al., 1999) or ADACOR (Leitao and Revisto, 2003). Each PH represents virtually a product that the FMS can manufacture. *Product Holons* are considered as virtual entities, which do not have a “physical” counterpart. In this sense, products do not have to be confused with parts which are the physical entities that are manufactured from a raw material. Every time a new manufacturing order of a product is scheduled in the system, the corresponding PH is notified, adding the required number

of new parts to its internal queue and starting their processing. *Product Holon* is responsible of the manufacturing of its parts so it has to dispatch the parts to the machines, track the information about their progression and notify upper layers when they are completed.

There are as many PHs as different products the system can manufacture, being all of them interconnected in a ring manner. This means that PHs can be dynamically added or removed from the system as it evolves. Additionally, every PH is linked, through a database, to a process plan modelled in STEP-NC which contains the information to manufacture it.

PH communicated directly between them to negotiate the assignation of idle resources based on the number of parts that each PH has to produce and their deadlines. This dispatching mechanism is done through a cyclic and circular mechanism in which the *Coordinator Holon* acts as referee. This dispatching mechanism is based on an heterarchical approach resulting in a highly resistant to disturbances process. Moreover, it allows dispatching the parts according to the real and instantaneous state of the system (taking into account available resources, tools or fixtures for example) maximizing the resource usage without needed to execute heavy and complex algorithms. For further details, refer to (Querol et al., 2015).

Additionally to communicating between them and with the *Coordinator Holon*, PHs also can communicate with all the machines of the system. After a PH successfully dispatches one of its parts to a machine for its processing, it must send the required part of the process plan (a setup typically) to the destination machine, so it can manufacture it. In the same way, when a machine finishes executing the setup of a part the corresponding PH is notified so it can dispatch the part to the next workstation.

From a technological point of view, all *Product holons* are modelled using the same Composite FB. This CFB contains a FBN composed of 14 FBs of different types which handle the communication with other holons, track the parts progression and negotiate the dispatching.

Machine Holon

Each *Machine Holon* represents a resource of the manufacturing system, such as a milling machine, a coordinate measure machine or an automated conveyor. For reasons of simplicity, only the case of machining resources, i.e. numerically controlled machines, will be presented here, since they are the main components of flexible manufacturing systems. However, other resources could be modelled in a similar manner.

Machine Holons are modelled as a FBN composed of three Composite FBs, belonging each of these CFBs to a different functional layer as depicted in

Author.

figure 4. This means that each MH can be further divided in three smaller components (inferior order holons) that will be referred as MH_L2, MH_L1 and MH_L0.

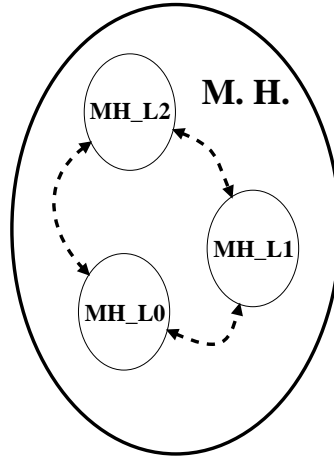


Figure 5: Machine holon internal architecture.

MH_L2 is in charge of tracking the state of the machine and its tools, communicating with other holons and, monitoring and storing the results of previous manufacturing operations execution. Once a part is dispatched to a machine, the MH_L2 receives the information corresponding to the setup that must be executed. The setup is then analysed and if alternative routes are detected, the best one is chosen according to the real state of the machine. These setup alternatives often include tool selection in function of availability or wear or operation sequence in function of mounted fixtures. MH_L1 main functionality is translating (post-processing) the STEP-NC models into machine code that underlying CNCs can interpret. This post-processing stage is required since most of current generation of CNCs cannot handle complex modelling languages like STEP-NC, only accepting standardized or proprietary G-Code (ISO6983) languages. MH_L1 additionally also applies adaptive correctors which modify the operation toolpaths based on the previous results of similar operations, tools wear or fixture offsets. These corrective actions, together with process plans with alternatives, enhance the autonomy and robustness of the manufacturing control since the own machines can take their own decision to adapt the manufacturing process of a part based on its environment (tools, fixtures or historical data). At this point it must be noted, that STEP-NC plays a major role in these corrective actions because it allows encapsulating all the design and process planning information into structured information models that the control can understand and exploit. This could not be possible if

traditional techniques were used to model process plans and manufacturing operations.

Finally, MH_L0 CFB acts as a communication interface between the MH and the physical device. The format of this communication varies from a device to another and is strongly related to the grade of openness of the own device. For PC-based CNCs, this communication is usually carried out through shared objects, like Microsoft COM technology. For CNCs with dedicated hardware and/or software serial or Ethernet interfaces can be used. Aside of the used interface, the most important feature of this communication with the CNCs is the accessible data. Controller devices data is usually accessed through some kind of predefined Application Programming Interface (API), depending of the openness of the controller this API will be more or less complete. Some APIs allow retrieving all the data of the control device enabling the much richer control systems since more adaptive actions can be made. Others however are more closed and only allow accessing a limited set of variables like machine current state or loaded manufacturing programs.

The *Machine Holon* model proposed has two important features that must be noted. On the one hand, since it is composed of three separated and independent holons, also modelled using IEC61499 FBs, MH can be distributed between different devices. This distribution may be useful to optimize the number of control devices and its distribution. Additionally, it can be crucial when using legacy devices which may need dedicated communication interfaces. On the other hand, the proposed MH can be easily reconfigured, since each of its components can be replaced without affecting the others. For example, if a machine controller is updated and it changes the physical interface, a new MH_L0 can be developed to communicate with the new controller, remaining the MH_L2 and MH_L1 untouched. In the same way, when developing the MH of a new machine, some of the inner holons can be reused. For instance, when adding a new machine of known type, like a milling machine, only the MH_L0 must be adapted while MH_L2 and MH_L1 can be directly reused from other designs.

Coordinator Holon

The *Coordinator Holon* is modelled as a single CFB and it is unique in the entire control architecture. It can be seen as a middle actor between *Machine Holons* and *Product Holons* which principal purpose is to collect and centralize information about the machines state so the PH can access it in an easier manner. Whenever a machine joins the system, leaves it or

Author.

changes modes (idle, error, busy) the *Coordinator Holon* is notified and updates consequently the state of the machines in the system.

CH also is in charge of periodically launching the circular dispatching mechanism in which PH negotiate to assign their parts to the available resources. Moreover, it acts as referee verifying the validity of the dispatch before it is produced.

Auxiliary function block networks

Aside of the above introduced holons, the designed architecture also is composed of several auxiliary FBN which cannot be classified as holons but that must be introduced to fully understand the proposed control. These FBN are mainly in charge of two tasks: orders scheduling and information routing through the holons. The orders scheduling currently is done entirely by a single CFB which register the incoming orders and tracks its execution progression. Orders are therefore directly scheduled once they are received in the system. This simplification has been done in this first stage of the research project since its main objective was to serve as prove-of-concept experience of utilization of FBs into HMS. However, in future extensions of the architecture, orders will be modelled as holons in a similar manner to Product Holons. This will allow implementing heterarchical and distributed scheduling algorithms which will be able to utilize the system information to optimize the production.

The other auxiliary FBN, the information router, is not a simplification but a requirement. To communicate *Product Holons* with *Machine Holons*, a routing mechanism must be implemented since, as will be detailed later, function blocks do not have any kind of framework to facilitate the communication between holons like other technologies do (Agents). Both MH and PH can be added or removed from the system at runtime and therefore proper discovery mechanism must be implemented and dynamic communication routes must be used.

5. Issues on utilization of function blocks for modeling and implementing high-level control tasks

On the one hand, function blocks intrinsically ease the development of holonic architectures since the FB itself could be considered as an holon according to the basic holon attributes. FBs as holons have a recursive structure, that is, FBs can be made of other FBs. They can also be considered autonomous entities since they can perform their own actions independently of their environment. Moreover, they can be connected in networks through well-defined interfaces allowing them to cooperate. Finally, they enable creating highly reconfigurable and reusable architectures as expected from

the holons. Because of this similitude between function blocks and holons, FBs can be used to model very complex control task such as scheduling, dispatching or monitoring using holonic principles. This allows applying new control strategies to implement control tasks like using distributed or cooperative algorithms which, as introduced earlier, can greatly improve the performance and robustness of a manufacturing system. Additionally, IEC61499 FBs enable developing highly distributable and reconfigurable control systems which are platform-independent since they are executed through runtimes, enhancing the reutilization of designs.

On the other hand, the function blocks have also shown important disadvantages when used to model high-level control tasks. One of the major problems found is their inability to handle complex information. FBs are conceived to execute simple algorithms programmed in sequential languages (like structured text) using primitive data types. When it is required to use more complex programming techniques such as Object-Oriented languages or structured and/or personalized data types they present serious limitations, having in many cases to exploit the use of SIFBs to by-pass them. Presenting a real case found in the proposed architecture, all the manufacturing information, including process plans, operations or geometries, were modelled using STEP-NC. These STEP-NC models are conceived using an objected-oriented approach, and therefore when inserted into the control software they must be defined well as pure objects or as structures of data. Since IEC61499 function blocks do not support using this kind of complex data types, SIFBs and serialization techniques had to be used to the information could be transmitted between the function blocks. This problem gets even worse when advanced reasoning techniques based on ontologies, easily implementable using Agents, want to be used to improve the communication and cooperation between holons.

Another important disadvantage was handling the communication between *Product Holons* and *Machine Holons*. As specified earlier, both PH and MH can be dynamically added or removed from the system, this prohibits using predefined static communication routes. To surpass this issue auxiliary FBNs dedicated to create and maintain these communication routes must be developed. When using Agents, this communication problem can be easily addressed since most of the MAS development environments already provide a framework with communication services like yellow and white pages functionalities.

On balance, IEC61499 function blocks ease the development of control architectures based on holonic principles allowing using more complex control strategies. Additionally, since it is a standard familiar to automation engineers, these can play a more relevant role in the development of the

Author.

manufacturing control, typically done by software engineers, improving the integration between low-level and high-level control tasks. At the same time, it has been proven that the usage of FBs greatly limits the use of complex data, making it harder to develop a manufacturing control that exploits complex product information like STEP-NC process plans. Moreover, the lack of global directives and services, even though improves the reconfigurability and componentization of the control, complicates the usage of intelligent and cooperative entities as they need a common framework and methods which eases their communication and cooperation. An option to increase the viability of function blocks on high-level control tasks could be to extend the function block model so it could handle complex information in an easier manner and integrate the FBs in a reference framework so they could cooperate. The idea of adapting the function block model so it better fits domain specific requirements has already been proposed by several authors, like Cantero (2015) where an adapted FB model with real-time constraints is used to develop the CNC controller of a glass cutting machine.

6. Conclusion and future works

In this work a control architecture that fulfils part of the requirements of next generation of manufacturing systems has been proposed, in which the holonic manufacturing principles are applied and IEC61499 function blocks technology is exclusively used. The proposed architecture, which has been designed and implemented for a flexible manufacturing system, has proven to greatly enhance the reconfigurability of the system. Additionally, it also improved the flexibility and adaptability of the manufacturing system by better integrating process planning information using STEP-NC models. Finally, the proposed architecture has shown an improved robustness to the disturbances thanks to its heterarchical and distributed approach.

However, in this research the limitations of using function blocks to model high-level control tasks has also been highlighted. FBs allow easily designing and implanting holonic architectures with a very high grade of reconfigurability and distribution but are strongly limited by their inability to handle complex information and their lack of a global framework to ease the communication between entities (holons for instance).

Having these limitations of function blocks in mind, future research projects will deal with designing an architecture with both Agents and FBs. The formers will handle the abstract reasoning and communication while FBs will be used as a middle layer between Agents and low-level controllers.

Design and implementation of a function block based holonic control architecture for a new generation flexible manufacturing system

References

Balasubramanian, S., Brennan, R. W., Norrie, D. H. (2001) 'An architecture for metamorphic control of holonic manufacturing systems' in *Comput.Ind.*, Vol. 46 pp. 13–31.

Brennan, R. (2007) 'Toward Real-Time Distributed Intelligent Control: A Survey of Research Themes and Applications' in *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 37 pp. 744-765.

Brussel, H. V., Bongaerts, L., Wyns, J., Valckenaers, P., and Ginderachter, T. (1999) 'A Conceptual Framework for Holonic Manufacturing Systems: Identification of Manufacturing Holons' in *Journal of Manufacturing Systems*, Vol. 18 No 1 pp.35–52.

Cantero, C. C. (2015) *Modelos y Plataforma IEC 61499 adaptados al control distribuidos de máquinas herramientas en sistemas de fabricación ágil*. PhD thesis, Universidad de Zaragoza, Zaragoza, Spain.

Christensen, J. H. (2002) 'IEC 61499 architecture, engineering methodologies and software tools' in *Proceedings of 5th IEEE/IFIP International Conference*, pp. 221–228.

Christensen, J. H. (2003) 'HMS/FB architecture and its implementation' in *Agent-Based Manufacturing: Advances in the Holonic Approach*, pp. 53–88.

Fletcher, M., Garcia-Herreros, E., Christensen, J. H., Deen, S., Mittmann, R. (2000) 'An Open Architecture for Holonic Cooperation and Autonomy' in *Proceeding of Holo-MAS'2000*, pp. 224 – 230.

Giret, A., Botti, V. (2004) 'Holons and Agents' in *Journal of Intelligent Manufacturing*, Vol. 15 pp. 645-659.

HMS (1994) PR. HMS Requirements. <http://hms.ifw.uni-hannover.de/>: HMS Server

International Electrotechnical Commission (2005) International Standard IEC61499-1, *Function Blocks - Part 1 Architecture*, First edition. Geneva.

Author.

Koren, Y., Hu, S.J., Peihua, G., Shpitalni, M. (2013) 'Open-architecture products' in *CIRP Annals - Manufacturing Technology*, Vol. 62 pp. 719-729.

Leitao, P., Restivo, F. (2003) 'Identification of ADACOR Holons for Manufacturing Control' in *Proceedings of the 7th IFAC Workshop on Intelligent Manufacturing Systems*, pp 109–114.

Lewis, R (2001) *Modelling control systems using IEC61499*. 1sted. The Institution of Electrical Engineers, London.

Mehrabi, M. G., Ulsoy, A. G., Koren, Y., Heytler, P. (2002) 'Trends and perspectives in flexible and reconfigurable manufacturing systems' in *Journal of Intelligent Manufacturing*, Vol. 13 pp. 135-146.

Monostori, L., Kadar, B., Morel, G. (2003) (Eds.) 'Intelligent Manufacturing Systems' in *Proceedings of the 7th IFAC IMS'03 Workshop*.

Morel, G., Panetto, H., Zaremba, M.B., Mayer, F. (2003) 'Manufacturing Enterprise Control and Management System Engineering: paradigms and open issues' in *IFAC Annual Reviews in Control*. Vol. 27 No2 pp. 199-209.

Querol, E., Romero, F., Estruch, A.M., Serrano, J. (2015) 'Design of the Architecture of a Flexible Machining System using IEC61499 Function Blocks' in *Proceedings of the 6th Manufacturing Engineering Society International Conference*, Vol. 132 pp. 934–941.

Valckenaers, P. (1998) (Eds.) 'Special issue on Intelligent Manufacturing Systems' in *Computers in Industry*, Vol. 37 No. 3.

Valckenaers, P. (2000). (Eds.) 'Special issue on Intelligent Manufacturing Systems' in *Computers in Industry*, Vol. 43 No. 2.

Vernadat, F.B. (1996) *Enterprise Modelling And Integration, principles and applications*. Chapman & Hall, London.

Vrba, P.T., Marik, V., Hall, K.H., Staron, R.J., Maturana, F.P., Kadera, P. (2011) 'Rockwell Automation's Holonic and Multiagent Control Systems Compendium' in *IEEE Transactions on Systems, Man, and Cybernetics—Part C: Applications and Reviews*, Vol. 41, No. 1, 2011

Design and implementation of a function block based holonic control architecture for a new generation flexible manufacturing system

Vyatkin, V. (2013) 'Software Engineering in Factory and Energy Automation: State of the Art Review' in *IEEE Transactions on Industrial Informatics*, Vol. 9 No. 3 pp. 1234-1249.

Yoshikawa, H. (1995) 'Manufacturing and the 21st Century - Intelligent Manufacturing Systems and the Renaissance of the Manufacturing Industry' in *Technological Forecasting and Social Change*, Vol. 49, pp. 195-213.